REMARKS

Claims 1-56 remain pending in the application. Reconsideration is respectfully requested in light of the following remarks.

Section 103(a) Rejection:

The Office Action rejected claims 1-12, 15-27, 30-40, 43-53 and 56 under 35 U.S.C. § 103(a) as being unpatentable over Lee (U.S. Patent 6,345,276) in view of Tucker (U.S. Patent 6,178,519).

Regarding claim 1, the Examiner states that Lee teaches a method for managing a virtual heap for a process executing within a virtual machine executing within a device. Applicants respectfully disagree with the Examiner's characterization of Lee. In fact, Lee teaches a smart pointer for interrelating data structures in a shared memory heap (Lee, Column 10, lines 26-27) and that operates "in a multiprocessing virtual memory operating environment on a computer, ... which contains physical memory that is shared among the multiple processes executing on the computer" (Lee, column 5, lines 41-45). Furthermore, Lee illustrates in Figure 1 a computer 20 that includes a system memory 22, BIOS 26, operating system 35, and application programs 36, but fails to teach or illustrate a virtual machine. Additionally, Applicants can find no reference in Tucker that teaches or suggests a virtual machine.

Further Regarding claim 1, the Examiner contends that Lee teaches a method comprising "performing an atomic transaction on the virtual heap, wherein said performing the atomic transaction comprises performing one or more transaction tasks, and wherein said performing the atomic transaction changes a state of the virtual heap by modifying one or more portions of the virtual heap." Applicants respectfully disagree with the Examiner's interpretation of Lee. Lee in fact fails to teach performing an atomic transaction on the virtual heap, wherein said performing the atomic transaction comprises performing one or more transaction tasks.

At the Examiner's cited reference (Lee, column 5, lines 40-67), Lee discusses a computer system with physical memory that is shared among multiple processes (Lee, column 5, lines 43-45). Lee further discusses multiple processes loading files into a shared physical memory and the use of smart pointers when addressing portions of physical memory that are mapped to portions of the processes' virtual memory address space. Applicants can find no reference or mention in Lee regarding performing an atomic transaction on the virtual heap wherein said performing the atomic transaction comprises performing one or more transaction tasks, as the examiner contends. Additionally applicants can find no reference in Lee teaching changing a state of the virtual heap. In fact, Lee fails to teach that a virtual heap may have a state that can change. Lee teaches only that the state of a *file* may be stored in a heap (Lee, column 1, lines 24-32).

Further regarding claim 1, the Examiner states that Lee teaches "committing the atomic transaction by accepting the modification to the one or more portions of the virtual hea[p] if the one or more transaction tasks in the atomic transaction are performed without generating an error." As shown above, Lee fails to teach performing an atomic transaction and therefore must necessarily fail to teach committing the atomic transaction. Applicants can find no reference in Lee regarding any error generation or regarding determining whether transaction tasks are performed without generating an error.

Additionally regarding claim 1, the Examiner contends that Tucker teaches "rejecting the atomic transaction by restoring the virtual heap to the state of the virtual heap prior to said performing the atomic transaction if one or more of the one or more transaction tasks in the atomic transaction generates an error when performed." While Tucker does teach a database server that handles rollback for aborted applications (Tucker, column 5, lines 19-20, and Figure 2), Tucker fails to teach such a rollback feature for managing a virtual heap for a process executing within a virtual machine.

Furthermore, as shown above, both Lee and Tucker fail to teach a virtual machine. Therefore, Applicants assert that neither Lee, nor Tucker, nor any combination of Lee and Tucker teaches the limitations of claim 1 while managing a virtual heap for a process executing within a virtual machine.

For at least the reasons given above, the rejection of claim 1 is not supported by the prior art at removal of the rejection is respectfully requested. Similar remarks as discussed above in regard to claim 1 apply to claim 44.

Regarding claim 16, the Examiner contends that Lee teaches "providing an application programming interface (API) for performing heap operations on the virtual heap, wherein the API comprises functions for performing operations on portions of the virtual heap, and wherein the functions in the API are callable by processes executing within the virtual machine." Applicants strongly disagree with the Examiner's interpretation of Lee. The examiner supports his erroneous contention by citing a totally irrelevant passage in Lee describing hard disk drives, magnetic disk drives, and optical disk drives connected to a system bus by respective interfaces (Lee, column 4, lines 29-32). Applicants point out that the interfaces Lee is referring to are physical connections that have nothing whatsoever to do with providing an API for performing heap operations on a virtual heap as the Examiner suggests. Furthermore, Applicants can find no reference in Lee teaching or suggesting the use of an API for any purpose whatsoever.

Further regarding claim 16, Applicants assert that Lee fails to teach a second process calling a first function from the API to perform a first operation on a first portion of the virtual heap and also fails to teach performing the first operation on a first portion of the virtual heap in response to the second process calling the first function. As shown above, Lee fails to teach providing an API for performing heap operations on the virtual heap and therefore necessarily fails to teach a process calling a function from the API to perform an operation and must also fail to teach performing the operation in response to the process calling the function.

Additionally, Applicants can find no teaching in Lee regarding "committing the first operation on the first portion of the virtual heap by accepting the modifications to the first portion of the virtual heap if the first operation is performed without generating an error." As shown above, regarding claim 1, Lee fails to teach anything regarding detecting whether operations are performed without generating errors.

Further regarding claim 16, the Examiner states that Tucker teaches rejecting the first operation [performed in response to a process calling a function in an API for performing heap operations on the virtual heap] on the first portion of the virtual heap by restoring the virtual heap to the state of the virtual heap prior to said performing the first operation if the first operation generates an error when performed. Applicants disagree with the Examiner. Tucker teaches a database server that handles rollback for aborted applications (Tucker, column 5, lines 19-20, and Figure 2). Applicants can find no teaching in Tucker providing an API comprising functions for performing operations on the virtual heap and also fails to teach rejecting an operation performed in response to a process calling a function in the API if the operation generates an error when performed. In fact, Tucker teaches away from such an API by describing how "data is accessed directly" (Tucker, column 5, lines 24-25) and how processes read data "stored in shared virtual address space 26 as if [it] were their own transient memory" (Lee, column 5, lines 31-33).

For at least the reasons given above, the rejection of claim 16 is not supported by the prior art at removal of the rejection is respectfully requested.

Regarding claim 31, the Examiner asserts that Lee teaches a system comprising a device configured to execute a virtual machine, wherein the virtual machine is configured to execute a process. Applicants disagree with the Examiner's interpretation of Lee. Lee in fact fails to teach a device configured to execute a virtual machine. Lee teaches a smart pointer for interrelating data structures in a shared memory heap (Lee, Column 10, lines 26-27) and that operates "in a multiprocessing virtual memory operating environment on a computer, ... which contains physical memory that is shared among the

multiple processes executing on the computer" (Lee, column 5, lines 41-45). Furthermore, Lee illustrates in Figure 1 a computer 20 that includes a system memory 22, BIOS 26, operating system 35, and application programs 36, but fails to teach or illustrate a virtual machine. Additionally, Applicants can find no reference in Tucker that teaches a virtual machine.

Furthermore, as shown above, Lee fails to teach wherein the device is configured to perform operations on the virtual heap according to an application programming interface (API), and wherein the API comprises functions for performing operations on the virtual heap, and wherein the functions in the API are callable by the process. As argued above, regarding claims 1 and 16, Applicants can find no teaching or reference in Lee regarding any application programming interface, and in particular can find no mention of an API comprising functions for performing operations on the virtual heap that are callable by the process. Applicants point out that the Examiner's cite passage (Lee, column 4, lines 29-32) is referring to are physical connections between storage devices and a computer system which have nothing whatsoever to do with providing an application programming interface (API) for performing operations on a virtual heap as the Examiner suggests.

Given that Lee fails to teach an API for performing operations on the virtual heap, as shown above, Lee therefore must necessarily fail to teach an API configured to: perform or commit an atomic transaction on the virtual heap, as the Examiner contends. Not only does Lee fail to teach an API performing or committing atomic transactions comprising one or more transaction tasks on the virtual heap, but as argued above regarding claims 1 and 16, Lee fails to teach any type of performing or committing of atomic transactions.

Further regarding claim 31, the Examiner states that Tucker teaches an API comprising functions for performing operations on the virtual heap configured to reject the atomic transaction by restoring the virtual heap to the state of the virtual heap prior to the atomic transaction if one or more of the one or more transaction tasks in the atomic

transaction generates an error when performed. However, as shown above regarding claim 16, Tucker fails to teach an API comprising functions for performing operation on the virtual heap and therefore must necessarily fail to teach such an API configured to reject the atomic transaction by restoring the virtual heap to the state of the virtual heap prior to the atomic transaction if one or more of the one or more transaction tasks in the atomic transaction generates an error when performed.

For at least the reasons given above, the rejection of claim 31 is not supported by the prior art at removal of the rejection is respectfully requested.

Claims 13-14, 28-29, 41-42, and 54-55 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Lee in view of Tucker and further in view of Lawrence (U.S. Patent 6,629,113).

Applicants assert that claims 13-14, 28-29, 41-42, and 54-55 are dependent from claims that have been shown above to be independently patentable and therefore are themselves patentable.

Applicants also assert that numerous other ones of the dependent claims recite further distinctions over the cited art. Since the rejection has been shown to be unsupported for the independent claims, a further discussion in regard to the remaining dependent claims is not necessary at this time.

Information Disclosure Statements:

In regard to the information disclosure statements filed August 6, 2001 and August 16, 2001, the Examiner indicated that the information has not been considered. The Examiner also requested the applicants to identify relevant references. The art cited in these information disclosure statements was cited in several related patent applications owned by the same assignee as the present application in the same general field of technology. The references were cited to fulfill applications duty of candor under 37

CFR 1.56. No specific study of each reference in regard to the present claims has been performed. Thus, Applicants cannot comment as to the relative relevance of the cited references. Applicants also note that another information disclosure statement was filed September 17, 2001. Copies of the forms PTO-1449 from all three of these invention disclosure statements are included herewith for the Examiner's convenience along with the returned postcards showing that the IDSs and references were received by the Office. According to MPEP(III)(C)(2), "information contained in information statements which comply with both the content requirements of 37 CFR 1.98 and the requirements, based on the time of filing the statement, of 37 CFR 1.97 will be considered by the examiner." "Examiners must consider all citations in conformance with the rules and [MPEP 609]." Since the information disclosure statements are in conformance with 37 CFR 1.97 & 1.98 and MPEP 609, Applicants respectfully request the Examiner to "consider the documents in the same manner as other documents in Office search files are considered by the examiner while conducting a search of the prior art in a proper field of search", as required by MPEP 609.

CONCLUSION

Applicants submit the application is in condition for allowance, and notice to that effect is respectfully requested.

If any extension of time (under 37 C.F.R. § 1.136) is necessary to prevent the above referenced application from becoming abandoned, Applicants hereby petition for such extension. If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5181-46700/RCK.

Also enclosed herewith are the following items:

☒ Return Receipt Postcard

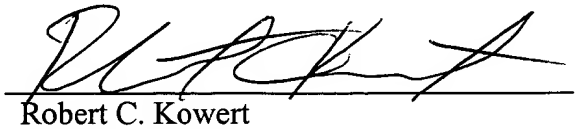
☐ Petition for Extension of Time

☒ Copies of previously submitted forms PTO-1449 and return receipt postcards

☐ Notice of Change of Address

☐ Fee Authorization Form authorizing a deposit account debit in the amount of $ for fees ( ).

Respectfully submitted,



Robert C. Kowert
Reg. No. 39,255
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8850

Date: April 15, 2004

#9
WM

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor(s): Abdelaziz et al
Serial No.: 09/587,076
Filing Date: 06/02/00
Title: DATABASE STORE FOR A VIRTUAL HEAP
Atty. Docket No. 5181-46700; P4453

The date stamp of the mail room of the U.S. Patent and Trademark Office hereon will acknowledge receipt of the attached 1) Information Disclosure Statement w/accompanying Form PTO-1449 and references B1 - B85; and 2) Return Postcard.

RCK\pwg Via First Class Mail date: Aug. 14, 2001

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor(s): Abdelaziz et al
Serial No.: 09/587,076
Filing Date: 06/02/00
Title: DATABASE STORE FOR A VIRTUAL HEAP
Atty. Docket No. 5181-46700; P4453

The date stamp of the mail room of the U.S. Patent and Trademark Office hereon will acknowledge receipt of the attached 1) Information Disclosure Statement w/accompanying Form PTO-1449 and references A1 - A119; and 2) Return Postcard.

RCK\pwg Via First Class Mail date: August 3, 2001

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor(s): Abdelaziz et al
Serial No.: 09/587,076
Filing Date: 06/02/00
Title: DATABASE STORE FOR A VIRTUAL HEAP
Atty. Docket No. 5181-46700; P4453

The date stamp of the mail room of the U.S. Patent and Trademark Office hereon will acknowledge receipt of the attached 1) Information Disclosure Statement w/accompanying Form PTO-1449 and references C1 – C43 & D1 - D9; and 2) Return Postcard.

RCK\pwg Via First Class Mail date: Sept. 13, 2001